Mini-Chapter — Modeling: Linear Regression

1. Why regression here? (Story over score)

Linear regression is a Swiss Army knife of modeling: fast to fit, easy to interpret, and a strong baseline for many financial problems. But its real superpower in this course is **storytelling**. You don't just get a score — you get a narrative about how the target moves with predictors, and the residuals tell you how your story fails.

2. Linear in coefficients (not necessarily in x)

"Linear regression" is defined by **linearity in coefficients (β)**. You can include transformed features like `x²`, `log(x)`, or even interactions like `x·z`. As long as the model is linear in β (i.e., β2 multiplies `x²` directly), you're still in linear regression territory.

Example:

- Notation: $y = β0 + β1·x + β2·x^2 + ε$
- Plot y vs x → curved.
- Plot y vs x^2 → straight. Takeaway: linearity is about the linear combination of features, not the raw shape when plotted versus one predictor.

3. The four core assumptions

1. **Linearity:** residuals should have no structure when plotted versus fitted values or predictors.
2. **Independence:** residuals shouldn't autocorrelate (a caveat in time series).
3. **Homoscedasticity:** constant residual variance; otherwise standard errors and inference distort.
4. **Normality:** residuals are normally distributed (important for small-sample inference; less crucial for point prediction with large n).

How to check quickly

- **Residuals vs fitted:** randomness is good; curvature is bad.
- **Histogram & QQ plot:** normality and tail behavior.
- **Residuals vs predictor:** exposes unmodeled nonlinearity.
- **Lag-1 residual plot:** informal independence check in time-indexed data.

4. Metrics: R^2 and RMSE (useful but not sufficient)

- **R^2** shows the fraction of variance explained but can be misleading if assumptions fail or if a spurious feature sneaks in.
- **RMSE** reports typical error in target units and is more comparable across models. Use metrics to compare models *after* you've visually verified assumptions.

5. Frequent misconceptions

- **"Higher R^2 means better."** Not necessarily. Check residuals first.
- **"Polynomial ⇒ nonlinear regression."** Not in the OLS sense; it's still linear in β.
- **"If sklearn runs, assumptions must be fine."** Code success ≠ statistical validity.

6. Financial-flavored example

Suppose daily **excess return** of an asset is driven by factors: market, size, value, momentum. A quadratic effect of momentum (convexity) and heteroscedastic noise proportional to market volatility are realistic touches. Fit a baseline model, then add `momentum²` and evaluate whether residuals improve.

7. Explanation vs prediction

- **Explanation:** Why does y move? Coefficient signs, magnitudes, and confidence intervals matter more; assumptions must be respected to avoid misleading inference.
- **Prediction:** What will y be? Cross-validated out-of-sample performance matters; assumptions still matter, but in different ways.

8. A minimal modeling workflow (assumptions → model → diagnostics → interpretation)

1. Hypothesize relationships (from EDA/FE).
2. Fit the simplest reasonable model.
3. Diagnose residuals (plots).
4. Iterate: add a targeted transformation (e.g., interaction or square).
5. Re-evaluate metrics *and* plots.
6. Document what changed and why.

9. What to write in your notebook

- Short paragraph for each assumption with a figure.
- A table with R^2 and RMSE for the baseline and improved models.
- A final paragraph: “Do I trust this model? For what purpose?”

10. What comes next

- If independence fails: consider time series models (AR terms) in Stage 10b.
- If variance changes with predictors: weighted least squares or variance modeling.
- If relationships are complex: feature engineering or regularization (Ridge/Lasso) in later stages.

Bottom line: Linear regression is your first serious conversation with the data. Let residuals speak — then decide what to do next.